REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
	Final Report	29 September 2001 - 29-Sep-04

4. TITLE AND SUBTITLE

Development of CRIPTE Code for Electromagnetic Coupling

5a. CONTRACT NUMBER
F61775-01-C0002

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Dr. Jean-Philippe Parmantier

5d. PROJECT NUMBER

5d. TASK NUMBER

5e. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Office National d'Etudes et de Recherches Arospatiales (ONERA)
2, Avenue Edouard Belin
Cedex 4
Toulouse 31055
France

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

EOARD
PSC 802 BOX 14
FPO 09499-0014

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
SPC 01-4034

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report results from a contract tasking Office National d'Etudes et de Recherches Arospatiales (ONERA) as follows: The contractor will investigate methods of updating computer modeling code developed by French researchers. This code was originally developed as part of an experiment performed under the joint US-France international data exchange program on the atmospheric electricity/aircraft interactions, DEA-AF-79-7336 (started 1979 and updated Mar 1997). ONERA will apply the CRIPTE code on the EM interaction on a full scale Mirage III aircraft wing to validate the code.

15. SUBJECT TERMS

EOARD, High Power Microwaves, Software, Electromagnetic pulses

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18, NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			MICHAEL KJ MILLIGAN, Lt Col, USAF
UNCLAS	UNCLAS	UNCLAS	UL	50	19b. TELEPHONE NUMBER *(Include area code)* +44 (0)20 7514 4955

ELECTROMAGNETICS AND RADAR DEPARTMENT

Final Report

Development of the CRIPTE Code for Electromagnetic Coupling – 3rd year final report

RF 6/06483 DEMR - January 2005

PHYSICS

ONERA

ELECTROMAGNETICS AND RADAR DEPARTMENT

Final Report N° RF 6/06483 DEMR

January 2005

Development of the CRIPTE Code for Electromagnetic Coupling
3rd year final report

Written by :

J.P. Parmantier, Head of Electromagnetic Compatibility and Detection Research Group

Approved by :

For J.-L. Boulay, Director
Electromagnetics and Radar Department
F. Christophe

This document comprises 49 pages

Centre de Toulouse
2, avenue Edouard Belin BP 4025 31055 TOULOUSE Cedex
Tél. : 05 62 25 25 25 - Fax : 05 62 25 25 50
Office National d'Études et de Recherches Aérospatiales SMP

SMP

IDENTIFICATION CARD of ONERA REPORT N° RF 6/06483 DEMR

Issued by : **ELECTROMAGNETICS AND RADAR DEPARTMENT**	Contracting Agency : **EOARD**	Contract Number : **F61775-01 - C-0004**
	Programme card number : **EMR 701**	Date : **January 2005**
Title : **Development of the CRIPTE Code for Electromagnetic Coupling - 3rd year final report**		
Author(s) : **J.P. Parmantier**		

SECURITY CLASSIFICATION : **de Défense**		Timing Classification Off	
Title	: **SANS MENTION DE PROTECTION**	Title	: **Sans objet**
ID card	: **SANS MENTION DE PROTECTION**	ID card	: **Sans objet**
Report	: **SANS MENTION DE PROTECTION**	Report	: **Sans objet**

Abstract :

This document describes the work carried out during the last 6-months of the 3rd year of the EOARD contract F61775-01-C002.

Ce document décrit le travail réalisé durant les 6 derniers mois de la troisième année de la commande de l'EOARD numéro F61775-01-C002.

Key words:

ELECTROMAGNETIC COUPLING ; ELECTROMAGNETIC COMPATIBILITY

COUPLAGE ELECTROMAGNETIQUE ; COMPATIBILITE ELECTROMAGNETIQUE

Centre de Toulouse
2, avenue Edouard Belin BP 4025 31055 TOULOUSE Cedex
Tél. : 05 62 25 25 25 - Fax : 05 62 25 25 50
Office National d'Études et de Recherches Aérospatiales

ONERA

SMP

DISTRIBUTION LIST of ONERA REPORT N° RF 6/06483 DEMR

Distribution of report

■ **Outside ONERA :**

EOARD
Contracting Office
223/231 Old Marylebone Road
London NM1 5th
UNITED KINGDOM — M. MILLIGAN, B. WANN ---------- 3 ex.

AFRL/DEHE
3550 Aberdeen Avenue
Kirtland Air Force Base
NM 87117
USA — J. YAKURA, W. PRATHER ---------- 3 ex.

■ **Inside ONERA :**

DEMR — Mmes BERTUOL, JUNQUA, MM. ALLIOT, BOULAY, CHRISTOPHE, MICHIELSEN, PARMANTIER ---------- 7 ex.

DCT/CID ---------- 1 ex.

Distribution of identification card only

■ **Outside ONERA :**

CEDOCAR ---------- 1 ex.

■ **Inside ONERA :**

DEMR/Palaiseau — C. LAURENT ---------- 1 ex.

Systematic Distribution : D, DSG, DAJ, DTG, DSAC, MM. les directeurs de MFE, PHY, MAS, TIS, GMT

Centre de Toulouse
2, avenue Edouard Belin BP 4025 31055 TOULOUSE Cedex
Tél. : 05 62 25 25 25 - Fax : 05 62 25 25 50
Office National d'Études et de Recherches Aérospatiales

ONERA

CONTENTS

1. INTRODUCTION 5

2. UPDATE ON THE DEVELOPMENT OF A USER'S INTERFACE FOR THE LAPLACE CODE 6

2.1. Objectives 6

2.2. Groups of contours 7

2.2.1. Main features 7

2.2.2. Main source code modifications 7

2.3. Material types 9

2.3.1. Main features 9

2.3.2. Main source code modifications 10

3. STRUCTURE OF THE CRIPTE 4.1 CDROM 14

3.1. Organization of the CDROM 14

3.2. Source files 15

3.2.1. Source files for CRIPTE editor 15

3.2.2. Source's files for CRIPTE calculation 15

3.2.3. "INCLUDE" files (.h) for CRIPTE editor and CRIPTE calculation 16

3.2.4. Files for the environment of CRIPTE editor 16

3.3. Installation of the CRIPTE Code 17

3.3.1. Subroutine for the verification of the computer supporting CRIPTE editor and CRIPTE calculation 17

3.3.2. The deadline date to run CRIPTE editor and CRIPTE calculation 17

3.3.3. Screen-configuration's file: « ecran.h » for CRIPTE editor 17

3.3.4. Printing files for CRIPTE editor 18

3.3.5. Makefiles: « Makefile_CRIPTE_Edit » and « Makefile_CRIPTE_Calcul » 19

3.3.6. XPM Libraries 20

3.3.7. XGRAB screen-capture software 20

3.3.8. In-Line Help of CRIPTE editor 20

3.3.9. Compilation 20

4. PROSPECTS FOR THE CONTINUATION OF THIS WORK 21

5. ANNEX: LAPLACE USER'S MANUAL 22

1. INTRODUCTION

This document is the final activity report of the contract F61775-01-C002 (third and last term, notified May 30th, 2002). It describes the work performed during the second half year of the third-term contract.

The different topics developed in the frame of this study are the following:

- Update of the development of a user's interface for the LAPLACE module inside the CRIPTE code,
- Installation of CRIPTE for MURI on RF effect partners,
- First delivery of the CRIPTE 4.1 version.

In addition, two papers have been published or accepted for publication, related to the work carried out within this grant:

- The paper entitled "Numerical Coupling Models for Complex Systems and Results", has been published in the IEEE Trans. on Electromagnetics, Volume 46, number 3, pp. 359, 367.
- The paper entitled "A network formulation of the Power Balance Approach for high frequency coupling", by I. Junqua, J-P. Parmantier and F. Issac, submitted to the Electromagnetics journal, has been accepted.

The structure of the document is the following:

- In the first section, we describe the current achievements on the development of LAPLACE's user's interface. The updated version of the available user's manual is presented in the Annexe 1,
- The second section describes the content of the CDROM containing the first delivery of the CRIPTE 4.1 version. This version contains the new version of the LAPLACE module,
- Finally the fourth section presents the possible prospects for the continuation of this grant.

2. UPDATE ON THE DEVELOPMENT OF A USER'S INTERFACE FOR THE LAPLACE CODE

2.1. Objectives

The work developed during the first 6 months of 2004 dealt with providing a user's interface to the LAPLACE program by integrating it in the CRIPTE_Editor environment. Doing so, there exist now an edition module and a calculation module as for CRIPTE. LAPLACE is now part of the CRIPTE code with its user's interface. The calculation module of LAPLACE can be called from the edition module or separately from CRIPTE.

We recall here that the objective of this work is not to develop a fancy interface as the one available in the commercial version of CRIPTE. The objective is that the so-called "research" version benefits of a simple interface since LAPLACE is one of the key software of the CRIPTE environment.

At mid term of this year, the computer version allowed the user to have exactly the same feature for editing the cross-section geometry of a multiconductor transmission line as the previous in-line version. The work developed in the second half year concerns the following developments:

- the definition of groups of contours,
- the management of materials with material types.

In addition, another constraint in the development dealt with insuring the compatibility of the ".geo" file format despite the introduction of the new features. This compatibility is very important in order to keep running the code with already existing ".geo" file.

The reader will refer to the user's manual which describes additional information on those group and material features.

The resulting version delivered at AFOSR, CRIPTE 4.1, still requires testing and validation which explains the different line comments appearing in the back window while the software is running[1][2]. Therefore, we consider that the reference version of CRIPTE is still CRIPT 4.0.

[1] In this version, the management of the horizontal menu has been modified in order to handle it as a subroutine that can be called from any subroutine. This type of implementation, which is currently available for the LAPLACE module only, will be generalized to the other modules of CRIPTE.

[2] In this version, the way to enter the language choice is different from the 4.0 version. Previously, this choice was made by answering a question. Now it is made my clicking the language button in the horizontal menu at the beginning of the program.

2.2. Groups of contours

2.2.1. Main features

Groups of contours have been introduced in order to carry out common actions on contours. Each contour may now belong to a group (even if it is not mandatory). However, it cannot belong to more than one group. The types of actions envisaged at this moment are:

- the destruction,
- the moving,
- the rotation,
- the modification of materials,
- ...

So far those actions are not implemented in the current version V4.1 of the CRIPTE.

This notion of contour is also important to manage macro geometry of models. We mainly think of dielectric coated wires, twisted pairs, three wire cables or random cable bundles.

So far, those macro models are not implemented in the current V.4 version of CRIPTE.

2.2.2. Main source code modifications

2.2.2.1. Variable declaration

The new parameters have been introduced in the file ***geo_laplace.h***:

integer ***maxgroup***: maximum number of groups
initialization: ***parameter(maxcontour=maxgroup)***
integer ***nb_group***: number of groups
character*40 ***nom_group(maxgroup)***: name of the group
integer: ***nb_cont_group(maxgroup)*** number of contours in the group
integer: ***liste_group(maxgroup,macontour)*** list of the contours in a group

All those variables are in the common named "group":

common/group/nom_group,liste_group,nb_cont_group,nb_group

2.2.2.2. Initialization of the variables

The initialization of the values of those variables is made in the ***init_cablinde*** module of **crmblnew_dessin_edit.f**.

nb_group =0
nom_group(maxgroup)=' '
nb_cont_group(maxgroup)=0
liste_group(maxgroup,macontour)=0

2.2.2.3. Introduction of the group information in info_laplace

The information on the groups has been introduced at two levels:

- when the contours are listed
- when all the groups are listed (at the end of the information)

2.2.2.4. Reading a geo file

When the contours are listed, we use a former blank line to specify the group to which the contour belongs. The used format is:

- *read(unitnum,'(1x,a8,a40)',err=420) nom_group_tmp* for reading the geo file (routine lecture_fichier_geometrie in crmblnew_dessin_edit.f)
- *write(unitnum,'(1x,"GROUP: ",a40)') nom_group(i)* for writing the geo file (routine *ecriture_fichier_geometrie* in crmblnew_dessin_edit.f)

Despite this new format, the old geo files can still be read.

2.2.2.5. List of the main routines and functions

Function *function groupe_existe(nom)* in crmblnew_dessin_edit.f .

This function returns the number identifier of the group (nom_group). If the name of group does not exist, the value is 0 (default value of the function).

Routine *verif_group(nom_group_tmp,i)* in crmblnew_dessin_edit.f .

This routine is called when the geo file is read. First, the program checks if the group **nom_group_tmp** exists. If not, this one is created only if the identification number of the contour **num_cont** does not belong to another group. If the contour belongs to another group, the conductor is not added to the **num_group** group list.

Function **cont_appartient_group(num_cont)**

Gives the number of the contour to which the number of contour **num_cont** belongs. If the contour does not exist, the contour number is set to 0.

Routine *mise_a_jour_contour(numcont,nbcont_tube)* in crmblnew_dessin_edit.f

Makes all the numbering again in order to have all the data stocked at the top of the tables.

This routine is called:

- when a group is deleted
- when a contour is deleted
- when a contour is removed from a group.

Routine *mise_a_jour_groupes_destruction_contour(numcont)* in crmblnew_dessin_edit.f

This routine relabels all the contours in the list_group table before calling *mise_a_jour_groupes* in the same routine.

In the *supprimer_contour_laplace* routine, it is checked that the if the removed contour was the only one in the list of contours of the group, this group is deleted.

2.3. Material types

2.3.1. Main features

2.3.1.1. General principles

Before, the relative dielectric of the right hand and the left hand sides of a contour was given explicitly by a real number. In the new version, the material is now identified by 5 parameters:

1. a name
2. a loss characteristic **lc**:
   a. a loss tangent, **tgδ** for a dielectric material or
   b. a conductivity **σ** for a metal material
3. the relative permittivity, **ε**, similar to the relative dielectric definition in the previous version.
4. the electric nature of the material (metal or dielectric)
   a. dielectric
   b. metal

The loss tangent is not used in the current version of the LAPLACE code but it will be used in future versions of the code (in order to calculate conductance matrices or the resistance matrices for instance). The important thing is that the actual calculation module of LAPLACE (Laplace3) can still be used with those new parameters (no modification required).

The materials will be defined in a special edition menu where the user will be able to create and modify materials.

2.3.1.2. Compatibility with the old geo files

In order to insure the compatibility with the old geo files, the list of the materials does not appear explicitely in the geo file but it is built in memory when reading the geo file.

Once read, the list of materials is accessible in the user's interface and the user may define or modify the materials at its own will.

The materials identified in a geo file are listed as "***material_n***" where "***n***" stands for the material number.

2.3.1.3. Metallic materials

A metallic material will always be characterized by:

- **ε = 0** ("abusive" value already at use in the former version of LAPLACE).
- lc = σ = finite constant real value

Besides, a perfect metallic material will be characterized by:

- lc = σ = infinite (real value to simulate infinite)

2.3.1.4. Dielectric materials

A dielectric material is characterized by:

- lc = tgδ= finite constant real value
- ε = finite value.

Particularly, a perfect dielectric material will be characterized by:

- lc = tgδ= zero (real value to simulate zero)
- ε = finite value.

Now, a material is defined for the surrounding **medium** in order to reduce any incoherence between left and right contours. Besides, the user is able to test this value in the future in order to check the coherence of the affectation of the materials when reading the geo file.

2.3.1.5. Definition of metallic and dielectric contours

We have chosen to impose the definition of the type of contour even if this one could have been deduced from the nature of the materials characterizing the left and right hand sides. Therefore:

- If the contour is declared to be a metal, the user is asked which hand side is the metal. Then, he will have to give the name of the dielectric material of the other side. No metallic material choice will be possible.
- If the contour is dielectric, the user will be asked to provide the material name of the left and the right contour.

2.3.2. Main source code modifications

2.3.2.1. Variable declaration

The following parameters are added in the **geo-laplace.h** file:

- *integer **max_material***: maximum number of materials allowed in the code
- *parameter (**max_material**=maxcontour)*: initialisation of the maximum number of materials allowed in the code
- *real **zero_material***: declaration of the value of a zero dielectric permittivity or loss characteristic. Under this value a dielectric permittivity will be considered as null.
- *parameter (**zero_material**=1e-8)*: initialization of the value of a zero sigma or zero loss tangent.
- *real **infinite_material***: declaration of the value of an infinite sigma or loss tangent. Over this value a conductivity or a loss tangent will be considered as infinite
- *parameter (**infinite_material**=1e8)*: initialization of the value of an infinite sigma or loss tangent.
- *character*8 **name_material (max_material)**:* declaration of the name of a material (similarly as contours and circuits are defined by names in LAPLACE)
- example for the material i:

  name_materia(1)= name of he material number i
- *integer **mat-type(max_material)***: defines the type of material
  - *mat-type=0*, dielectric material (loss characteristic=constant loss tangent and finite epsilon)
  - *mat-type=1*, metallic material (loss characteristic=sigma and epsilon=0)
- *integer **nb_material***: defines the total number of materials that have been defined

- *real* ***val_material(max_material,2)***: defines the value of the loss characteristic (conductivity or loss tangent) and permittivity
  example for the material i:
  val_material(1)= sigma (or loss tangent)
  val_material(2)= epsilon
- *real* ***val_material_medium (2)***: defines the value of the conductivity and permittivity of the surrounding medium
  example:
  val_material_medium (1)= sigma (or loss tangent) of the medium
  val_material_medium (2)= epsilon of the medium
- *integer* ***num_mat_cont(maxcontour,2)***: defines the number id of the material for each contour
  example for the contour i:
  num_mat_cont (i,1)= number id. of left material
  num_mat_cont (i,2)= number id. of right material

All those parameters are defined in the common ***material***:

common/material/ nb_material ,name_material, mat_type,val_material, type_mat_medium, val_material_medium, num_mat_cont

2.3.2.2. Initialization of the variables

The initialization of the values of those variables is made in the ***init_geometrie*** module of **crmblnew_dessin_edit.f**. The two first materials are always:

- the medium (dielectric type)
- the perfect metal (metallic type)

The initialization sequence is as follows:

```
nb_material=2
name_material(1)='medium'
name_material(2)='metal'
val_material_medium(1)=zero_material
val_material_medium(2)=1.0
type_mat_medium=0
val_material(1,1)= val_material_medium(1)
val_material(1,2)= val_material_medium(2)
type_mat(1)=type_mat_medium
val_material(2,1)= infinite_material
val_material(2,2)=zero_material
type_mat(2)=1
do i=2,max_material
val_material(i,1)=zero_material
      val_material(i,2)=1.0
      type_mat(i)=0
end do
do i=1,maxcontour
      num_mat_cont(i,1)=0 (initialisation left material)
      num_mat_cont(i,2)=0 (initialisation right material)
end do
```

2.3.2.3. Updating the material information when reading the geo file

The reading is made in the *lecture_fichier_geometrie* routine of the file crmblnew_dessin_edit.f

The information in the geo file is written at the line of the old comment on the epsilon right and left.

Now this line contains the name of the materials used under the format:
read(unit,'material left and right ',a8,' ',a8) name_mat_l,name_mat_r

The names *name_mat_l* and *name_mat_r* will be incremented in the memory (even if they are blank) while reading the values of epsilons and loss characteristics in the line just after.

The values of epsilon and loss characteristic appear in the line just after (previously, only the epsilon were given):
read(unit,) eps_l,eps_r,sigma_l,sigma_r,mat_type_l,mat_type_r*

The loading in memory of the *name_material, val_material* variables is made by testing the pairs of values *(sigma_l, epsilon_l)* and *(sigma_r,epsilon_r)* variables (see later in the document the routine *affect_sigma_epsilon_exist*).

Then, the *num_mat_cont* variable can be updated depending on the left or right hand side. This update is made by the following procedure where *numcont* is the number identification of the contour:

call affect_sigma_epsilon_exist (sigma_r, epsilon_r, name_mat_tmp_r, num_mat_r, mat_type_r)
num_mat_cont (i,2)= num_mat_r
call affect_sigma_epsilon_exist (sigma_l, epsilon_l, name_mat_tmp_l num_mat_l, mat_type_l)
num_mat_cont (i,1)= num_mat_l

2.3.2.4. Updating the materials when reading the geo file

Routine *affect_sigma_epsilon_exist (sigma,epsilon,name_mat_tmp,num_mat,type_mat_tmp)*
This routines first tests the pair (*sigma,epsilon*) real inputs and determines if this pair is already characterizing a material (see function *test_sigma_epsilon_exist)*.

```
num_mat= test_sigma_epsilon_exist(sigma,epsilon,type_mat_tmp)
if (num_mat.eq.0)

c test if the material does not exist
nb_material=nb_material+1
num_mat=nb_material+1
type_mat(num_mat)=type_mat_tmp
if (name_mat_tmp(1:1).ne.' ')
      name_material(num_mat)=name_mat_tmp
```

```
else
      call convertir_entier_chaine(num_mat,reponse,nl)
      name_material(num_mat)=material_//reponse(1:nl)
end if
else
c test if the material exists
if (name_mat_tmp(1:1).ne.' ')
      name_material(num_mat)=name_mat_tmp
else
      call convertir_entier_chaine(num_mat,reponse,nl,xxx)
      name_material(num_mat)=material_//reponse(1:nl)
end if
 end if
```

Function ***test_sigma_epsilon_exist(sigma,epsilon,type_mat_tmp)***

This function returns the identification number of the material to which belongs the set (***sigma,epsilon, type mat tmp***). By default, the value is null.

```
test_sigma_epsilon_exist=0
val_rel=1e-3
fini=0
i=1
do while (i.le. nb_material.and.fini.eq.0)
      sig_rel=(sigma- val_material(i,1)) - (sigma- val_material(i,1))
      eps_rel=(epsilon- val_material(i,2))/ (epsilon- val_material(i,2))
      type_mat_rel=(type_mat_tmp-type_mat(i))/type_mat(i)
      if (sig_rel.le.val_rel.and.eps_rel.le.val_rel.and.type_matrel.le.val_rel) then
            test_sigma_epsilon_exist=i
            fini=1
      end if
i=i+1
end do
```

2.3.2.5. Writing the geo file

The writing in the geo file consists in:

- mentioning the names of the right and left materials of the contour *i*:
  write(unitnum,'("Materiau gauche et droit: ",a8," ",a8)') ***name_material****(num_mat_cont (i,1)),* ***name_material****(num_mat_cont (i,2))*

- mentioning the values of sigma and epsilon for the left hand sides:
  write(unit,)* ***val_material****(num_mat_cont (i,1),2),* ***val_material****(num_mat_cont (i,2),2),* ***val_material****(num_mat_cont (i,1),2),* ***val_material****(num_mat_cont (i,2),1),****mat_type****(i)*

3. STRUCTURE OF THE CRIPTE 4.1 CDROM

3.1. Organization of the CDROM

This section describes the structure of the CDROM (*Fig. 1*) provided with this final report.

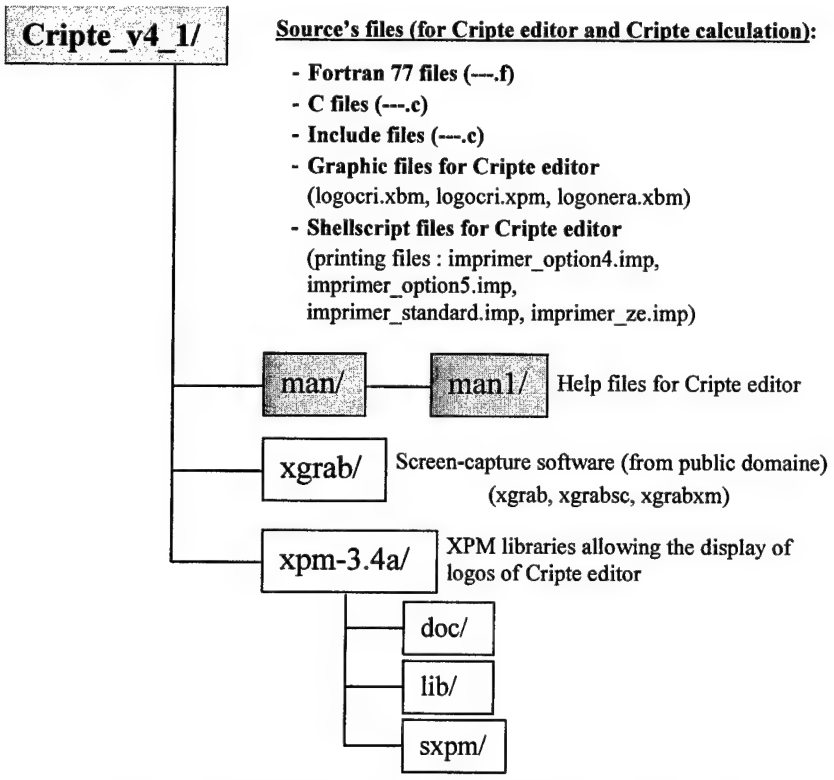

Fig. 1 - Structure of the CRIPTE 4.1 installation CDROM

For the installation of CRIPTE on a computer, we suggest to keep the same organization of the files on the computer.

3.2. Source files

In blue are mentioned the main source files having been modified for the programming of LAPLACE's user's interface.

The structure of the code is similar to CRIPTE 4.0.

3.2.1. Source files for CRIPTE editor

blt_batch.f	freq_edit.f	sauver_reseau_edit.f
cad_commun.f	gestion_run_lecture.f	sources_edit.f
cad_commun_edit.f	graf_edit.f	synoptique_edit.f
calcul_zy_edit.f	imprimer_edit.f	transfert_hpspice_edit.f
caljon_edit.f	interactif_edit.f	transfert_seq_dir_edit.f
callc_edit.f	interconnexion_edit.f	transfo_mesures_s_edit.f
creer_s_edit.f	jonction.f	util_tube_edit.f
cripte_edit.f	jonction_edit.f	**visu_tube_edit.f**
crmblnew_dessin_edit.f	jonction_ideale_edit.f	
crmblnew_edit.f	**laplace2_edit.f**	Hyper_edit.c
crtubegra_edit.f	lect_params.f	*(Hyper_LINUX_edit.c)*
date.f	lect_tubecrm_edit.f	*(Hyper_UNIX_edit.c)*
definir_res_par.f	libtub_edit.f	environ_cripte_edit.c
definir_res_par_edit.f	menu_ligne_edit.f	**garp_edit.c**
definition_de_s.f	**menu_outils_edit.f**	hman-cripte_edit.c
definition_type_s.f	module_rayonne_edit.f	idate.c
definition_ws.f	module_resultats_edit.f	imprimer_global_edit.c
definition_ws_edit.f	numerotation_compactage.f	q_s_edit.c
diago_edit.f	numerotation_opt.f	quick_sort_edit.c
dinvmat_edit.f	outils_calculation.f	swapf_edit.c
effectuer_numerotation.f	outils_j_ideale_edit.f	swapi_edit.c
effectuer_numerotation_edit.f	prises_de_terre_edit.f	verif_machine.c
excel_edit.f	rayonnement_edit.f	*(verif_machine_DEC.c)*
extraction_donnees_edit.f	reseau_edit.f	*(verif_machine_HP.c)*
f_gnup_edit.f	resol_sparse.f	*(verif_machine_LINUX.c)*
fonction_bessel.f	resol_sparse_edit.f	*(verif_machine_SUN.c)*
form_reseau.f	run_edit.f	*(verif_machine_ibm.c)*
form_reseau2.f	run_initialisation.f	(verif_machine_indy.c)

3.2.2. Source's files for CRIPTE calculation

alloc_dyn_calcul.f	definition_ws_calcul.f	resol_sparse_calcul.f
analytique_calcul.f	effectuer_numerotation.f	run_initialisation.f
blt_batch.f	fonction_bessel.f	sortie2_calcul.f
cad_commun.f	form_reseau.f	
cripte_calcul.f	form_reseau2.f	idate.c
date.f	gestion_run_calcul.f	verif_machine.c
def_tube_calcul.f	gestion_run_lecture.f	*(verif_machine_DEC.c)*
definir_res_par.f	jonction.f	*(verif_machine_HP.c)*
definir_res_par_calcul.f	lect_params.f	*(verif_machine_LINUX.c)*
definition_de_s.f	numerotation_compactage.f	*(verif_machine_SUN.c)*
definition_de_s_calcul.f	numerotation_opt.f	*(verif_machine_ibm.c)*
definition_type_s.f	outils_calculation.f	*(verif_machine_indy.c)*
definition_ws.f	resol_sparse.f	

3.2.3. "INCLUDE" files (.h) for CRIPTE editor and CRIPTE calculation

Hyper.h	matrice_s.h	run.h
HyperP.h	menu.h	second_membre_ws.h
appel_c.h	**menu_hor_lap_edit.h**	solution.h
caracteristiques.h	milieu.h	source1.h
compactage.h	module_mailleur.h	source2.h
ecran.h	module_resultats.h	source3.h
ecran_LINUX.h	noeud1.h	source4.h
ecran_UNIX.h	noeud2.h	source5.h
gamzc.h	numerotation_onde.h	source6.h
geo_laplace.h	onde.h	sous_reseau1.h
ground.h	onde_plane.h	sous_reseau2.h
imprimer.h	option_gra.h	table_couleur.h
jonction1.h	**parametres.h**	tube1.h
jonction2.h	petit_gamma.h	tube2.h
jonction3.h	position.h	tube3.h
jonction_ideale.h	reseau.h	tube4.h
jonction_ideale2.h	reseau2.h	tube_inter.h
lect_laplace.h	resolution1.h	zy.h
ligne_de_texte.h	resolution2.h	
	rlcg.h	

3.2.4. Files for the environment of CRIPTE editor

cripte	logocri.xbm	imprimer_option4.imp
	logocri.xpm	imprimer_option5.imp
	logonera.xbm	imprimer_standard.imp
		imprimer_ze.imp

3.3. Installation of the CRIPTE Code

3.3.1. Subroutine for the verification of the computer supporting CRIPTE editor and CRIPTE calculation

The subroutine for the verification of the computer called « verif_number() » is in the C-files « verif_machine_xxx.c » (xxx = « HP » or « DELL » or « SUN » or « ibm » or « LINUX », ...).

The identification number of the computer must be entered in the subroutine.
This number is unique for each machine. To determine this number:

- For a UNIX system:

There is a small C-language source file allowing the obtention this number depending on your computer

This file is "quelle_machine.c", the command of compilation of this file is:

```
cc –DSUN  quelle_machine.c  -o  quelle_machine    (for a SUN computer)
cc –DHP   quelle_machine.c  -o  quelle_machine    (for an HP computer)
cc –DIBM  quelle_machine.c  -o  quelle_machine    (for an IBM computer)
cc –DDEC  quelle_machine.c  -o  quelle_machine    (for a DEC computer)
```

- For a LINUX system:

Type the command: uname -a > text-file.txt

The identification number of the computer for a UNIX system, or the information contained in « text-file.txt »for a LINUX system, must be entered in the « verif_machine_xxx.c » C-files

3.3.2. The deadline date to run CRIPTE editor and CRIPTE calculation

The deadline date must be indicated in the subroutine called « entrer_date_limite » contained in the « verif_machine_xxx.c » C-files (xxx = « HP » or « DELL » or « SUN » or « ibm » or « LINUX », ...).

3.3.3. Screen-configuration's file: « ecran.h » for CRIPTE editor

Usually, the dimensions of the screen are automatically found for a UNIX system, and the widget of CRIPTE is automatically fitted to those dimensions. For a LINUX system, there are some mistakes, so there are two specific files for the screen-configuration:

- « ecran_UNIX.h » and
- « ecran_LINUX.h ».

It is necessary to copy one of these files in the file « ecran.h »

- For a UNIX system: cp ecran_UNIX.h ecran.h
- For a LINUX system: cp ecran_LINUX.h ecran.h

3.3.4. Printing files for CRIPTE editor

The printing files are specific to each computer (depending on the software installation ...), they use the free (from public domain) software « xgrab ».

• Ways to operate printing files

The subroutine « imprimer » contained in « imprimer_edit.f » offers different printing features:

1- Standard
2- Postscript (in a postscript file)
3- ZE+logo
4- Option4
5- Option5

- The buttons 1, 3, 4 and 5 call directly subroutines contained in the « imprimer_global_edit.c » C-file.
- Before calling a subroutine contained in the C-file « imprimer_global_edit.c », the button 2 builts the Unix (or Linux) command calling « xgrab », this command is sent to the subroutine of « imprimer_global_edit.c ».
- Then the different subroutines of « imprimer_global_edit.c » call the shell-script files « imprimer_standard.imp » for button 1, « imprimer_option4.imp »for button 4, « imprimer_option5.imp » for button 6. For button 3, the appropriate subroutine creates a tempory file called « xgrabsc.tmp » and calls « imprimer_ze.imp »

Those subroutines in « imprimer_global_edit.c » use the include file « imprimer,h » where the run-time of xgrab (« xgrabsc ») must be indicated with the absolute path. The absolute path of the shell-script files « xxx.imp » must also be indicated.

• Adapting printing files for the installation of CRIPTE editor

- « imprimer_edit.f »:
  search the line: « com= ..... » (twice) and put the absolute path for « xgrabsc »
  (i.e.: com= «/usr/local/xgrab/xgrabsc.....»)
- « imprimer.h »:
  search the line « #define M_XGRABSC » and put the absolute path for « xgrabsc »
  (i.e.: #define M_XGRABSC«/usr/local/xgrab/xgrabsc»)
  search the line « #define M_SHELLIMP » and put the absolute path for the « xxx.imp » files
  (i.e.: #define M_SHELLIMP«/usr/local/cripte»)
- « imprimer_option4.imp », « imprimer_option5.imp », « imprimer_standard.imp »
  give the absolute command for xgrabsc
  (i.e.: /usr/local/xgrab/xgrabsc -cps -reverse -output xgrabsc.tmp)
  give the Unix (or Linux) command used on the computer for printing the temporary file « xgrabsc.tmp » on the postscript printers connected to the computer
  (i.e.: lpr -Plpc1 -h xgrabsc.tmp)
- « imprimer_ze.imp »
  give the Unix (or Linux) command used on the computer for printing the temporary file « xgrabsc.tmp » on the postscript printers connected to the computer
  (i.e.: lpr -Plpc1 -h xgrabsc.tmp)

3.3.5. Makefiles: « Makefile_CRIPTE_Edit » and « Makefile_CRIPTE_Calcul »

Here are described some points which have to be modified, depending on the computer where CRIPTE is installed

The points noted with "*" are just for CRIPTE editor (« Makefile_CRIPTE_Edit »). The other points are for both CRIPTE editor (« Makefile_CRIPTE_Edit ») and CRIPTE calculation (« Makefile_CRIPTE_Calcul »).

Note that the Makefile of the version 4.1 (file "Makefile_CRIPTE_Edit") has been upgraded in order to account for the new source files related to Laplace v.0

- F77: the command for FORTRAN compiler on the computer
  [i.e.: g77 –fugly-complex (for a Linux system), f77 (for a Unix system) ...]

- CC: the command for C compiler on the computer
  [i.e.: gcc –g (for a Linux system), cc (for a Unix system) ...]

- * CINCLUDES: the path of includes (X11, Motif, ...)
  [i.e.: –I/usr/X11R6/include]

- LINK: the linker command
  [i.e.: g77 (for a Linux system), f77 (for a Unix system) ...]

- * LDLIB: the path of libraries (X11, Motif, ...)
  [i.e.: LDLIB = -L/usr/X11R6/lib -lXm -lXpm -lXt -lSM -lXp -lICE -lXext -lX11]
  (more examples are given in the makefile...)

- CPP: the pre-processor.
  On some computers the pre-processor is in /lib/ccs. Sometimes it must be declared.

- CPPOPT: options for the pre-processor
  [i.e.: -P]

- OPTIONS: special options for the computer (type of computer, size of the screen, ...)
  [i.e.: -D_768x1024_ -DLINUX -DWITH_XPM -DWITH_HMAN]
  (descriptions are given in the makefile)

- * HMAN: Sources used for the Help in CRIPTE editor
  (when the option -DWITH_HMAN is used)
  For a Unix system: HMAN = hman_cripte_edit.c Hyper_UNIX.c
  For a Linux system: HMAN = hman_cripte_edit.c Hyper_LINUX.c

- DEBUG: choice of debugger
  [i.e.: -g –w]

- OU.SUIS.JE: the extension of the name of the run-time version of CRIPTE editor
  (for example, the name of the computer the date, or something else ...)
  [i.e.: edit]

- VERIFF: the name of the C file which verify the inner number of the computer and the deadline date

  [i.e.: verif_machine_LINUX.c]

- * IMPRIME: the name of the C file where are the printing subroutines

  [i.e.: imprimer_global_edit.c]

- EXEC: the name of the run-time version of CRIPTE editor

  [i.e.: EXEC = cripte$(OU.SUIS.JE)

- REPERTOIRE.CRIPTE: The absolute path of the directory where CRIPTE is installed

  [i.e.: /usr/local/cripte]

- REPERTOIRE.HELP: The absolute path of the directory where the help files of CRIPTE editor are installed

  [i.e.: /usr/local/cripte/man]

- * XPM.INCLUDES: The absolute path of includes for XPM

  [i.e.: /usr/X11R6/include/X11]

- * XPM.LIB: The name of the library for XPM

  [i.e.: /usr/X11R6/lib/libXpm.a]

3.3.6. XPM Libraries

To display the different logos, CRIPTE editor needs XPM Libraries. On most of computers those libraries are installed with the operating system (Linux for example), but on some computers (old operating systems...), those libraries are not available. So, the directory xpm-3.4a contains all the files to build those libraries (make file...)

3.3.7. XGRAB screen-capture software

As we explain in § 3.3.4, CRIPTE editor uses the screen-capture software XGRAB before printing. This software can be found in the public domain (for Linux systems for example), but a run-time version of XGRAB (« xgrabsc ») for SUN operating systems is given in the directory xgrab. In this directory, all the files for creating a new run-time according to the right operating system can be found.

3.3.8. In-Line Help of CRIPTE editor

CRIPTE editor has an « In-Line Help ». In order to use it, it's necessary to choose the option « WITH_HMAN » in the Makefile (see § 3.3.5). The help files (xxx.1) must be in the directory man/man1 as shown in *Fig. 1*.

3.3.9. Compilation

When all the previous points have been verified, the compilation for creating the run-time versions of CRIPTE editor and CRIPTE calculation is made by those two commands:

```
make –f Makefile_CRIPTE_Edit
make –f Makefile_CRIPTE_Calcul
```

4. PROSPECTS FOR THE CONTINUATION OF THIS WORK

In October 2004, a proposal for the continuation of this work has been sent to AFOSR. In this proposal, we propose several ideas for continuing the work to be carried out in parallel to the extension of the MURI on RF effects. The proposed work deals with the 3 main items described hereafter:

1 – CRIPTE delivery and support to the MURI group

The updates of the CRIPTE code will be regularly delivered to AFRL. In addition, always under AFRL's control, the code will be installed on computers of the MURI partners who ask for the code.

So far the following recipients are identified for continuation:

- - **AFRL** in Albuquerque (Point of contact: Dr. Joe Yakura)
- - **University of Missouri** (Point of contact: Prof. Naz Islam)
- - **ElectroScience Lab at the Ohio State University** (Point of contact: Prof. John Volakis)
- - **University of Illinois** (Point of contact: Prof. Giorgio Uslenghi)

2 – CRIPTE improvement

This work focuses on CRIPTE improvements that are useful for the kind of analysis performed in the RF-effects MURI program:

- **Graphical interface of the LAPLACE module**: enhancement of the current version developed in the first grant. The objective is to introduce the selection of contours with the mouse (problem of all partners concerned in modeling cables),
- **Automatic labeling of junction ports**: this feature is very important in the case of large junctions. Typically, this is what happened if junctions are calculated at aperture interfaces in a decomposition of domain approach (a research topic for the University of Illinois personnel),
- **Introduction of the Taylor-source model**: this addresses the means to add distributed currents calculated from detailed 3D Maxwell solvers. This feature will be useful for all the partners who are interested in the calculation of field-to-transmission lines (University of Missouri and the University of Ohio),
- **Review of the English user's manual** prepared by University of Missouri.

3 – Continuation of the BLT-oriented research on Power Balance Approach

The work started during the first grant will continue. New ideas to be investigated are the following:

- **additional models** for coupling cross sections (particularly cables)
- **methodology** for applying this approach to a complex system (especially to take into account of the real cavity effects)
- introduction of **norm concepts** (used by Dr. C.E. Baum in the theory of interference control)

5. ANNEX: LAPLACE USER'S MANUAL

LAPLACE_ MENU

General description

Coming from:	Called menus:
OUTILS-TUBES(1) ("LAPLACE" button)	No main menu called
Last update	December 2004

Table of contents

1 INTRODUCTION 25

2 DEFINITIONS 26
- 2.1 Description of the materials 26
  - 2.1.1 Material characteristics 26
  - 2.1.2 Material values in the ".geo" files 26
- 2.2 Types of contours 27
- 2.3 Description of the geometry of circular contours 28
- 2.4 Description of the geometry of polygonal contours 28
- 2.5 Description of the nature of a contour 28
- 2.6 Affectation of the unknowns 29
- 2.7 Definition of groups of contours 30
- 2.8 Description of circuits 30

3 DESCRIPTION OF THE MENUS 31
- 3.1 Main menu 31
- 3.2 General features 32
  - 3.2.1 Reserved names 32
  - 3.2.2 The « Info on geometry » menu 32
  - 3.2.3 LAPLACE's horizontal menu 33
  - 3.2.4 The display window 34
  - 3.2.5 The"Keyboard/Mouse selection" menu 35
- 3.3 « Create-geometry » menu 36
  - 3.3.1 The « Edit materials » menu 36
  - 3.3.2 The « Read geometry » button 38
  - 3.3.3 The « Create contour » button 39
    - *3.3.3.1 The "Circular contour" sub-menu 41*
    - *3.3.3.2 The "Polygonal contour" sub-menu 41*
  - 3.3.4 The « modify contour » button 42
  - 3.3.5 The « Delete contour » button 42
  - 3.3.6 The « Move contour » button 42
  - 3.3.7 The « Edit group » button 42
  - 3.3.8 The « Move group » button 43

3.3.9 The « Macro-Models » button 43
3.3.10 The « Create circuits » button 46
3.3.11 The « Delete circuit » button 47
3.3.12 The « Save geometry » button 47
3.4 The "Calculation" menu 47
3.4.1 Preliminary checks 47
3.4.2 Characterization of the calculation 48
4 SEE ALSO 49

1 INTRODUCTION

LAPLACE (graphical version v0) is a tool calculating the per-unit length inductance (L) and capacitance (C) electrical parameters of a transmission line from a model describing its cross-section geometry. For this, the Laplace equation is solved in two dimensions on this geometry. The numerical method used is a method of moments.

Compared to the former versions of the LAPLACE package, the new LAPLACE offers new features. Some of them are directly applicable in the current program. Others prepare future features of the software[1].

This menu offers two main capabilities:

- the description of the **cross-section geometry** of the cable with a user's interface based on a graphical library similar to the one used within CRIPTE. This feature comes from the in-line versions known under the names "*Laplace*" or "*Laplace2*" formally available with the CRIPTE package. The geometry is read and stored under a "**.geo**" format,
- the **calculation** of the per-unit-length parameters of the transmission line. This calculation is based on the former "*Laplace3*" program. Throughin the graphical menu, this program is launched automatically. The parameters are classically saved with a "**.tub**" and "**.t**" format (see FORMAT-TUBE(1)).

The geometry is supposed to be made of wires located over or inside a surface. The model accounts for a dielectric environment made of air, cable coating. However, the geometry does not account for several cable shield levels. For this capability, the user will have to look at the menu devoted to the creation of shielded cables (see CABLE-BLINDE(1)).

From a direct formulation consisting in determining the distribution of charges as a function of the electric potential on the contours, the LAPLACE menu calculates the capacitance, and inductance matrices. The charge distribution is supposed to be uniform on the conductors. The resulting file may be completed using different tools in the tube tool menu (the create-tube menu (see CREER-TUBE(1)) for instance). This happens when the user wants to introduce the resistance, R, or the conductance, G, matrix.

All the inputs of the menu are made in the **MKSA** international system.

The new features that have been added to the former version in the user's interface are the following:

- the definition of **materials**, indicated with a name. Now the materials are sorted in **dielectric materials** and **metallic materials**. In addition to the former epsilon value, a real part is added equal to a **conductivity** if the material is conductive or a **loss tangent** if the material is a dielectric.
- the definition of **groups** of contours allowing the user to carry out global operations on groups of contours. The group is also the key notion to define macro-models of cables, that is to say specific groups such as dielectric coated wire, two coated wire pairs, three coated wire cables, metallic ground plane,

[1] The features that are not already available in the software are indicated in blue in this document. They stand for the specifications of future developments in the code.

- the definition of open contours (that are virtually closed in order to be compatible with the general construction rule of the software),
- the perfectly conductive and infinite ground plane.

The new version of the calculation version is the following:

- the former *laplace3* calculation module can be directly called from the graphical user's interface,
- the calculation handles open contours,
- conductivity matrices can be calculated from the geometry and the material characteristics,
- resistance matrices can be calculated from the geometry and the material characteristics,
- perfectly metallic ground plane configurations can be handled in the calculation

The ".geo" file has been modified but is still compatible with the former versions of LAPLACE which means that old ".geo" files can be treated with the new version.

2 DEFINITIONS

2.1 Description of the materials

2.1.1 Material characteristics

The materials definition is a new feature of the software. Materials are now identified by:

- a name,
- a nature type:
  - **dielectric**
  - **metallic**
- a set of two constant real values:
  - a value for the **relative dielectric constant**
  - a **loss value**
    - if the nature of the material is dielectric, the loss value is equal to a **loss tangent**
    - if the nature of the material is conductor, the loss value is equal to **a conductivity**

Two types of materials are defined by default:

- **medium**, of dielectric nature, initialized to Epsilon =1 and loss tangent =0
- **metal**, of metallic nature, initialized to Epsilon =0 and sigma =infinite value. This infinite value is set to 10^8 (MKSA) in the code. Abusively, the program affects a permittivity equal to "0" to a metallic material.

2.1.2 Material values in the ".geo" files

In the new geo file format, the names of the materials and their nature are automatically saved in the geo file when giving the material properties of the left and right contours. The format for saving the values is the following:

```
chaine='Nature electrique'
```

```
  nl=longueur_chaine(chaine)
  write(unitnum,'(a)') chaine(1:nl)
  write(unitnum,*) nat_elec(i)
  write(unitnum,'(''Materiau gauche et droit: '',a8,'' '',a8)')
$name_material(num_mat_cont(i,1)),name_material(num_mat_cont(i,2))
  write(unitnum,*)
$val_material(num_mat_cont(i,1),2),
$val_material(num_mat_cont(i,2),2),
$val_material(num_mat_cont(i,1),1),
$val_material(num_mat_cont(i,2),1),
$type_mat(num_mat_cont(i,1)),
$type_mat(num_mat_cont(i,2))
```

When reading the old geo file format, the names are updated when reading the pair of materials values for the left and right hand sides of the contours:

- If the value of the dielectric constant is equal to zero, the nature of the material is automatically detected as metallic,
- If the value of the dielectric constant is not equal to zero, the nature of the material is automatically detected as dielectric.

2.2 Types of contours

The geometry is made of **closed contours**. The LAPLACE menu offers two possible types of geometry configurations:

- **circular contours**: this feature is generally used for wire descriptions (dielectric insulator or metallic parts),
- **polygonal contours**: this feature is generally used to model ground planes, reference surfaces, cable shields.

Each contour is characterized by a **name of contour**.

The contours are not supposed to touch each other. However a dielectric may surround a conductor. In that case, the dielectric must not touch other dielectric or conductor contours. Nevertheless, very small distances between contours may be considered simulating touching contours.

All the contours are oriented. The convention adopted for all the contours is that a contour (polygonal or circular) is always **counter clockwise** oriented (Figure 1) with respect to a normal to the window outgoing of the screen.

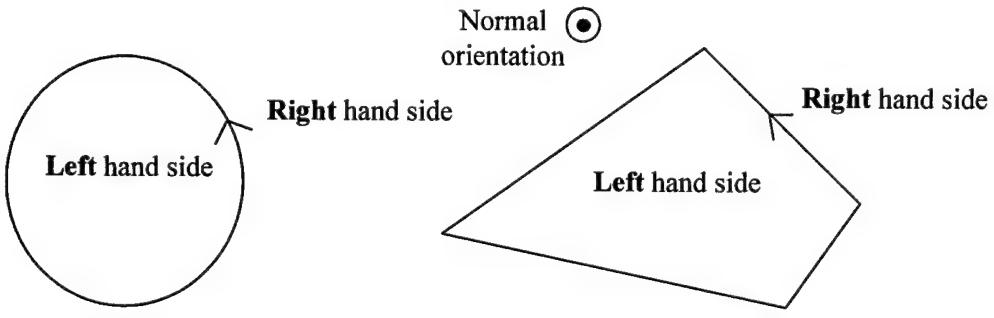

Figure 1: definitions of left and right hand sides with respect to the contours

2.3 Description of the geometry of circular contours

A circular contour is defined by:

- its **radius**,
- the x and y coordinates of its **center**.

2.4 Description of the geometry of polygonal contours

A polygonal contour is made of segments. For each polygonal contour a **number of segments** is given. Each segment is characterized by the ***x* and *y* coordinates** of its origin. Automatically, the pair of origin x and y coordinates of the following segment are the x and y coordinates of the remote end of the first segment. Consequently, in order to keep developing closed contours, the rule is that the remote x and y coordinates of the last segment are the origin coordinates of the first segment.

In order to be consistent with the definition of the orientation of the contours, the following rule is imposed for the construction of polygonal contours. Polygonal contours must be entered counter clockwise (Figure 2). If not, the program automatically changes the orientation.

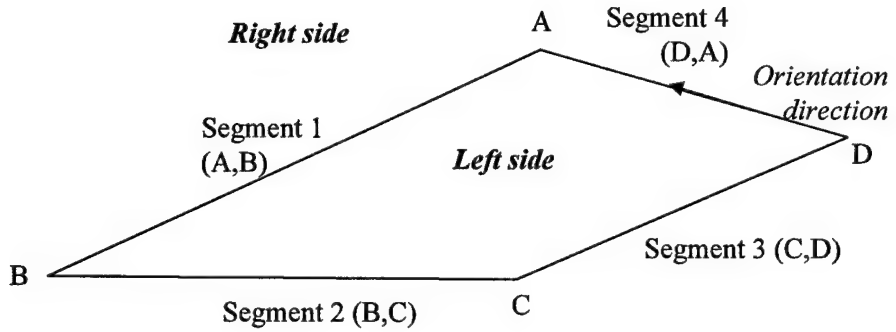

Figure 2: Example of a polygonal contour made of 4 segments

The reorientation of the polygonal contours is automatically made counter clockwise each time the segments are not in the appropriate orientation:

- when **reading** a ".geo" file,
- when **entering** a polygonal contour through the user's interface,
- when **saving** the ".geo" file.

2.5 Description of the nature of a contour

The contours may be of two types:

- dielectric (there are also called "**contrasts**"). In the resulting ".geo" file, the type of a dielectric contour is equal to "**0**".
- metallic (they are called "**conductors**" or "**frontiers**" as well). In the resulting ".geo" file, the type of a conductor contour is equal to "**1**".

Each contour always separates two dielectric regions characterized by two materials. The definitions of a left hand side and a right hand side are made with respect to the counter clock wise orientation of the contour.

The nature of a contour is characterized by:

- a **left hand side material** that contains the relative permittivity (simply called left permittivity later on) and the loss value (also called left loss value later on),
- a **right hand side material** that contains the relative permittivity (simply called right permittivity later on) and the loss value (also called right loss value later on).

A **metallic contour** (also called conductor) is supposed to have one of its two left or right hand sides full of metal. Therefore:

- if a conductor is the reference of an open cable geometry (that is to say, if it does not contain any dielectric or metallic contour), its **left material must be of metallic nature** (left permittivity is equal to "0"),
- if a conductor is a shield, that is to say, the reference of a closed geometry (which means it may contain dielectric or metallic contours), its **right material must be metallic** (right permittivity is equal to "0").

Consequently, a **contrast** will be simultaneously characterized by a left dielectric material and a right dielectric material (left non-zero relative permittivity and right non-zero relative permittivity). There is no check in the program to control if the permittivity and loss values are consistent with the permittivity values of the other contours. However, color being displayed for the dielectric values when drawing the geometry should help the user to detect possible errors (see farther the correspondence of the colors).

2.6 Affectation of the unknowns

On each contour, the unknowns are applied in the following way:

- For each circular contour, with a number N of **basis functions** de composed in N/2 **even** basis functions and N/2 **odd** basis functions.
- Along each segment of a polygonal contour, with a **number N of x and y abscissas** on which the basis functions are defined.

On each segment of a polygonal contour, different manners to distribute the unknowns can be chosen. The following options are available:

- **LINEAR**: linear distribution,
- **BEGIN**: concentration at the beginning of the segment
- **END**: concentration at the end of the segment
- **BEGIN_END**: concentration at the beginning and the end of the segment

If the number of abscissas is set to zero on a contour, the **contour is virtually opened**. This feature is also useful for modeling a finite dimension ground reference. For example, if this ground is a plane, the segments under the circular contours can have zero abscissas.

There is no real rule to determine the number of functions or the positions of unknowns in the case of a polygonal contour. The only precaution that might be given is that the number of unknowns in direct sight between two contours must be of the same order.

For coated wire descriptions, the experience proves that the following default values generally give good results:

- 20 basis functions on a frontier,
- 30 basis functions on a contrast around a frontier.

2.7 Definition of groups of contours

Several contours may belong to the same group. This is particularly important when global operations have to be applied on several contours (moving a group, deleting a group ...).

This notion is also important for the creation of macro models of geometry configurations as coated wires.

A group is characterized by:
- a name,
- a list of contours belonging to this group.

In the geo file, the fact that a contour belongs to a group is now indicted as follows before the list of information on the contour:

```
write(unitnum,'(1x,''GROUP: '',a40)') nom_group(num_group)
```

2.8 Description of circuits

The description of circuits is essential for the derivation of the L and C matrices. A circuit is made of a set of two metallic contours (or conductors or frontiers):
- one is a negative pole (also called "**minus**" pole),
- one is the positive pole (also called "**plus**" pole).

If N conductors are defined in the geometry, the user will not be able to define more than N-1 circuits. In the transmission line model, the circuits define the way the voltages are defined in the transmission line model, voltages being defined between the minus pole and the plus pole.

Generally, in usual transmission line models used in CRIPTE, all the voltages are referenced to the same and unique reference conductor. The model is then called a "**one-reference**" model. Therefore, one conductor is the same minus pole for all the circuits. Generally, this conductor is a surface (plane, surrounding shield ...).

If different minus poles are considered, the model is called a "**multiple-reference**" model. In that case, the user will have to be very prudent in the way he connects the resulting tube to other tubes when he builds the CRIPTE network corresponding to this transmission line.

Such reference notions are also encountered in shielded cable models (see CABLE-BLINDE(1)).

3 DESCRIPTION OF THE MENUS

In this section, we present the different menus and recall the identification menu number used is the ***menu_outils_edit.f*** source code file.

3.1 Main menu

The main menu (menu number 59) is the following (Figure 3):

- **Create Geometry**: this button allows the user to create the cross section geometry of its cable and create a « .geo » file.
- **Calculation**: this button allows the user to calculate the R, L, G and C matrices from a « .geo » file and create a « .tub » file and a « .t » file.
- **Exit**: this menu exits the LAPLACE program.

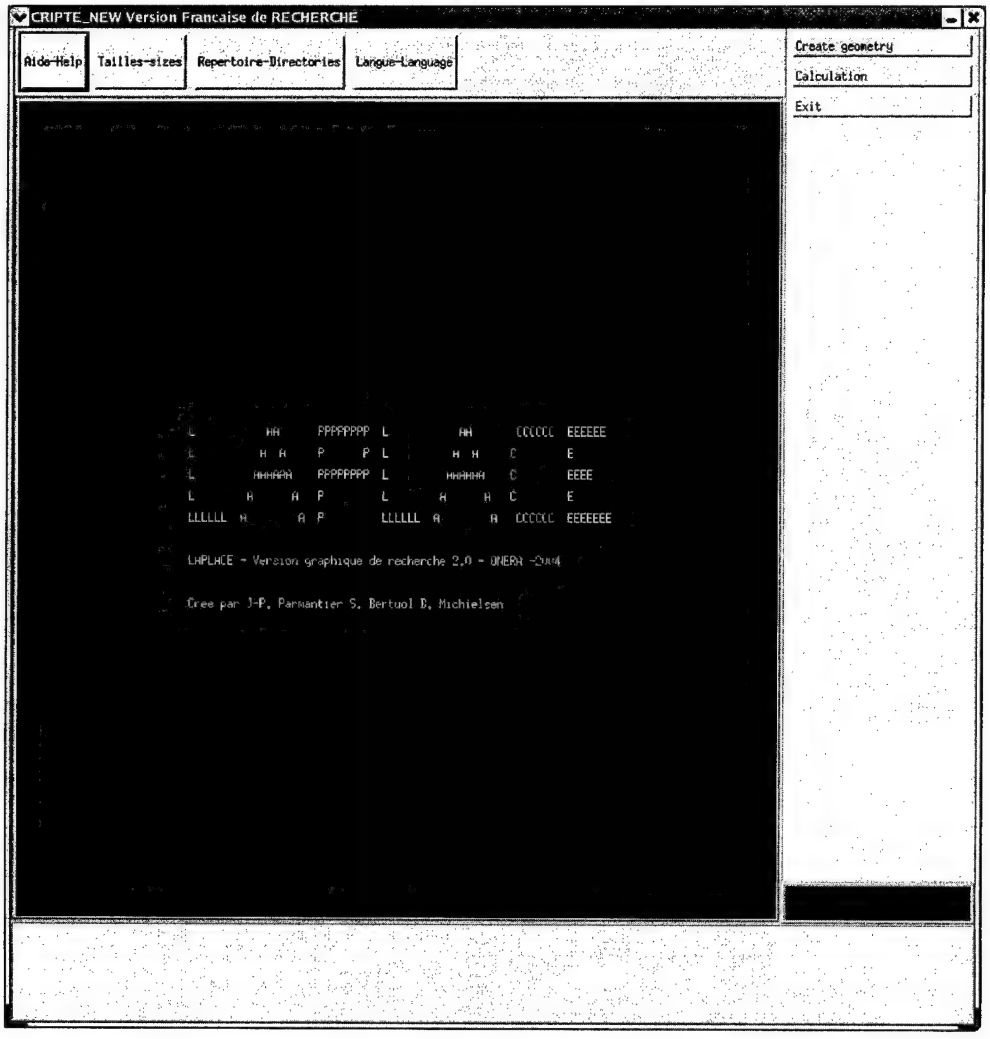

Figure 3: Main window of the LAPLACE V0 user's interface

3.2 General features

3.2.1 Reserved names

The following beginning of names are reserved since they can be automatically affected to a contour or a circuit by the program:

- « **cond_**», for a contour name,
- « **diel_**» for a contour name,
- « **cir_**», for a circuit name,
- « **mat_**», for a material name.

In addition, blank names used to specify a contour name or a circuit name are not allowed.

Such automatic naming and labeling may occur in the following situations:

- when reading a « .geo » file, if names of contours or circuits in this file are similar to names of contours already available in memory. In that case, the name of contours, or circuits are called "cond_n" or "diel_n" or "cir_n", where « n » will be equal to the computer number of the contour or the circuit. Similarly, the names of materials can be called "mat_k", where "k" stands for a material number.
- when generating an automatic cross-section geometry from a list describing the geometry.

However, those reserved names can be used when reading a « .geo » file.

3.2.2 The « Info on geometry » menu

All the sub-menus under the « **Create geometry** » button have a « **Info on geometry** » button. All these buttons display the same information on the geometry currently available in memory (Figure 4).

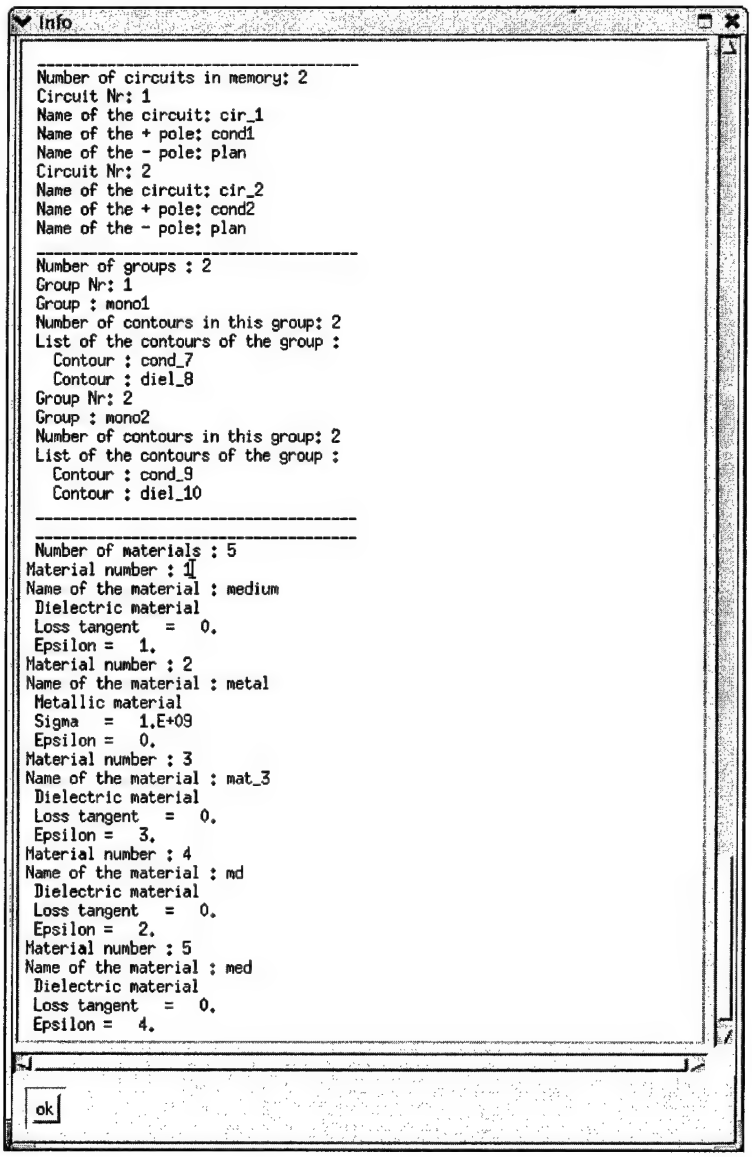

Figure 4: Example of information on material types in the Information menu

3.2.3 LAPLACE's horizontal menu

LAPLACE has the same horizontal menu (see Figure 5). The following options are available:

1 – **Help**: this button launches the in-line help,

2 – **Sizes**: this button reminds the user the sizes of the different tables used in the module,

3 – **Info**: this button provides some information on the object selected with the mouse,

4 – **Directory**: this button offers the user the possibility to change the default directories,

5 – **Zoom**: this button allows the zoom in the drawing window. As usual in CRIPTE, the user has to imagine that the zoomed window is in a rectangle for which he has to click in order to specify the upper left corner and the lower right corner of the rectangle,

6 – **Back**: this button goes back to the previously zoomed window, until there is no more zoomed window available,

7 – **Redraw**: this button redraws the display of the cross section geometry. This capability is sometimes useful to refresh the display,

8 – **Print**: this button goes to the current print menu (see MENU-HORIZONTAL(1))

9 – **Display**: this menu allows the user to select the type of information he wants to display in the display window. A click on this button launches the following vertical menu:

- **Contour name**, to specify if he wants to display the names of the contours (default value is [y] (yes)).
- **Circuit name**, to specify if he wants to display the names of the circuits (default value is [y] (yes)).
- **Group belonging,** to specify if the name of the Group has to appear on the display,
- **Color attributes,** to specify if the colors and display is based on the dielectric or loss (conductivity and loss tangent) values,
- **Positions of unknowns** on contours if he wants to display the position of the unknowns (default value is [n] (no)). Such a display is generally useful in zoomed windows when the user wants to control the balance of the unknown distribution between different contours.
- **Return**: when the display specification is finished, the program goes back to the previous window

3.2.4 The display window

Figure 5 gives an example of a display window in LAPLACE V0. The reader will recognize wires with circular contours, some wires being without dielectric coating other being with dielectric coating. The figure also displays a polygonal contour (one describing a ground plane).

The color displayed corresponds to the relative dielectric permittivity range. The following main colors are associated to the following main permittivity values:

- 0 = Epsilon < 1: black,
- 1 = Epsilon < 2: white
- 2 = Epsilon < 3: red
- 3 = Epsilon < 4: green
- 4 = Epsilon < 5: dark blue
- 5 = Epsilon < 6: pink
- 6 = Epsilon < 7: light blue
- 7 = Epsilon < 8: kaki green
- 8 = Epsilon < 9: medium blue
- 9 = Epsilon < 10: faded green

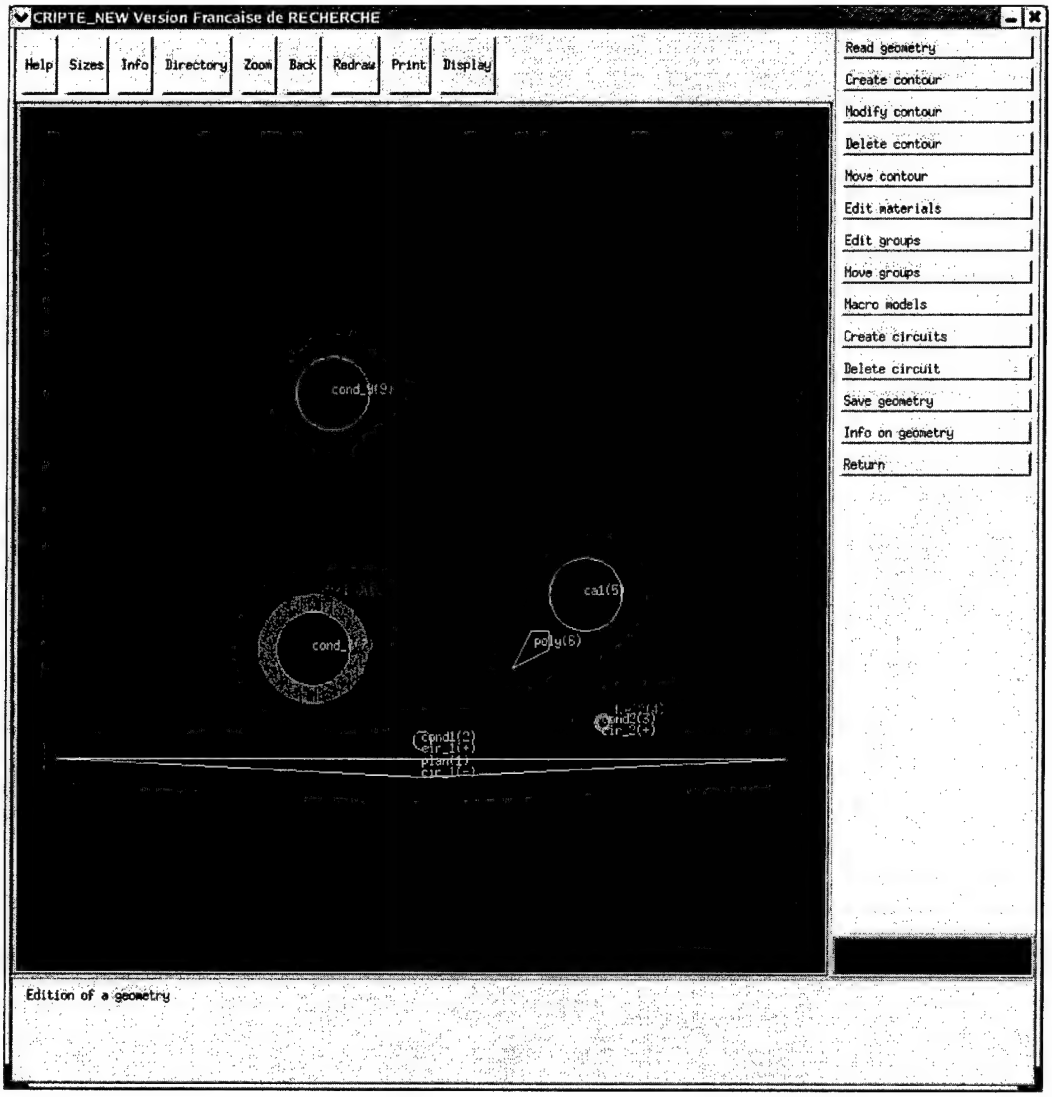

Figure 5: Example of LAPLACE's display window

3.2.5 The"Keyboard/Mouse selection" menu

This vertical menu appears at different places inside the program (menu number 66). It offers the following options (Figure 6):

- **With the keyboard**; the information has to be answered with the keyboard by answering the question caption at the bottom of the main window. This feature is sometimes necessary since the selection of a contour with the mouse is not possible or ambiguous,
- **With the mouse** ; the information is answered by clicking the display window with the mouse,
- **Return**: goes back to the previous menu.

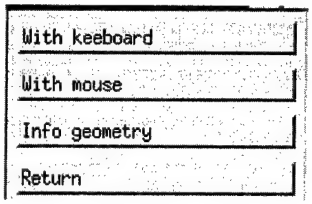

Figure 6: The « Keyboad/Mouse » selection menu

3.3 « Create-geometry » menu

This menu offers the following options (menu number 60):

1. **Read geometry**
2. **Create contour**
3. **Modify contour**
4. **Delete contour**
5. **Move contour**
6. **Edit materials**
7. **Edit groups**
8. **Move groups**
9. **Macro models**
10. **Create circuits**
11. **Delete circuit**
12. **Save geometry**
13. **Info on geometry**
14. **Return**

3.3.1 The « Edit materials » menu

This menu offers the user the capability to edit some materials characteristics. The menu offers the following options (Figure 10, menu number 74):

1. **Modify medium**
2. **Create material**
3. **Modify material**
4. **Delete material**
5. **Info materials**
6. **Info geometry**

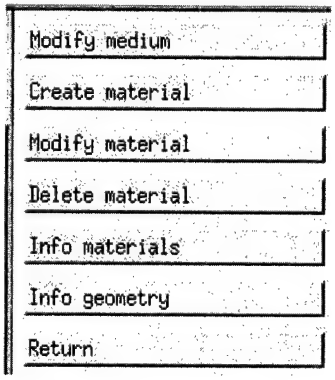

Figure 7: The « material » edition menu

- **Modify medium** allows the user to characterize the dielectric and loss values of the medium, this one being necessarily of dielectric type,
- **Modify medium, Create material** and **Modify material** allows the user respectively:
  - to characterize the dielectric and loss values of the medium,
  - to create a new material. For this, the user has to enter the name of a material that does not exist yet,
  - to modify an existing material. For this, the user has to provide the name of an already existing material.

    For those 3 menus, the user has to indicate the electrical nature of the material (Dielectric or Metal). For this, he his asked the following question:
    '***Dielectric (d) or metallic (m) material ? [D]***', the default value being dielectric [D].

    - If the material is of metallic nature, the user is directed to the following menu for which the loss value is equal to a conductivity, sigma (menu number 75):
      1. **Sigma**
      2. **Info geometry**
      3. **Return**
    - If the material is of dielectric nature, the user is directed to the following menu for which the loss value is equal to a loss tangent, sigma (menu number 76):
      1. **Loss tangent**
      2. **Info geometry**
      3. **Return**

Figure 8 gives an example of those edition menus in the case of the dielectric material.

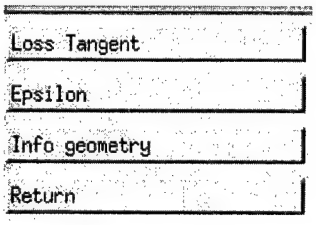

Figure 8: The « electric characterization of material » menu (dielectric case)

- **Delete material** allows the user to remove a material from the list of materials. For this, the user has to provide the name of an existing material,
- **Info in materials** allows the user to display the list of all available materials (see Figure 9). This menu appears in several menus.
- **Info on geometry** is the usual displays of information window.

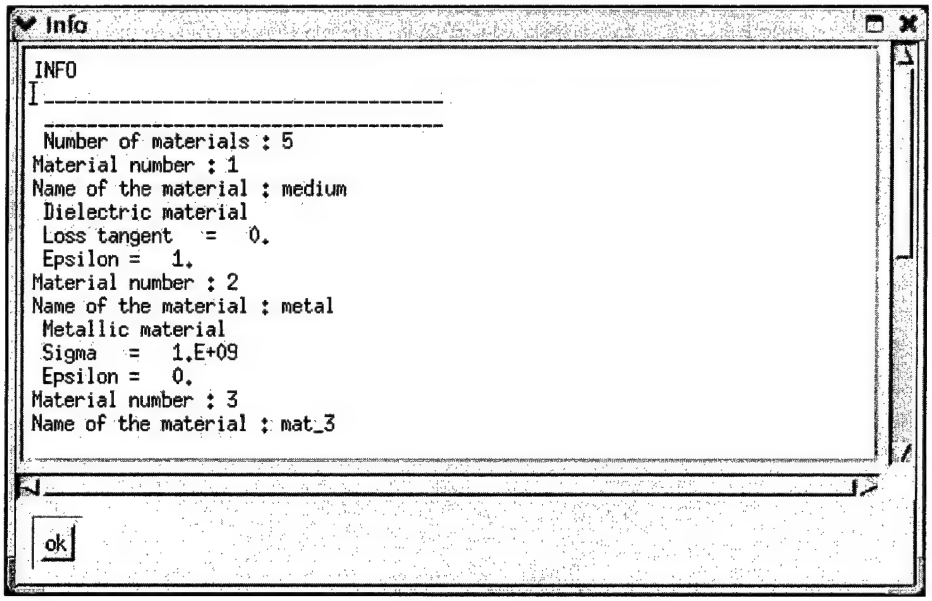

Figure 9:« Information on materials » window

3.3.2 The « Read geometry » button

This button offers the capability to load in memory a geometry stored in a « .geo » file.

When clicking this button, the reader has to select a name of an existing « .geo ». The selection is made in the usual window for selecting a file.

The contours are renamed if the names are already used for available contours in memory:

- ***"cond_n"*** is the new name given to the frontier, where « n » is the computer id of the current contour,
- ***"diel_n"*** is the new name given to a contrast, where « n » is the computer identifier of the current contour.

If those contours appear to be also circuit poles, the names of the poles are automatically changed.

Circuits are deleted in the following situations:

- if one of the contour names affected to the minus or plus poles does not exist,
- if the names of the plus and minus poles are identical.

Circuits are renamed if the names are already used for available circuits in memory: ***"cir_n"*** is the new name given to the circuit, where « n » is the computer identifier of the current contour.

The **list of materials** is updated with respect to the names of materials mentioned in the ".geo" file or the pairs of values (loss value and dielectric value). If no material name is given, the name of materials are labeled as ***"mat_n"*** where "*n*" stands for the currant available material number.

The **group list** is updated if group names are mentioned in the ".geo" file.

3.3.3 The « Create contour » button

This button is designed in order to create a new contour and load it in memory.

First, the user has to enter the name of the contour he wants to create. Several possibilities may occur (Figure 10):

- the name of the contour already exists in memory. No creation is possible.
- the name of the contour does not exist, the programs goes to the following menu (menu number 61):
  1. **Geometry contour**
  2. **Electric nature**
  3. **Info on materials**
  4. **Info on geometry**
  5. **Return**

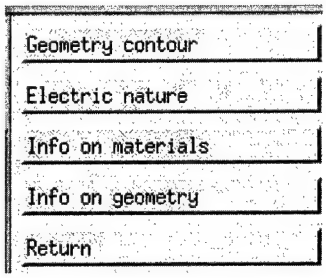

Figure 10: The « contour » edition menu

- **Geometry of the contour** allows the user to define if he wants a circular or a polygonal contour, with the following menu (Figure 11, menu number 79):
  1. **Circular contour**
  2. **Polygonal contour**
  3. **Info geometry**
  4. **Return**

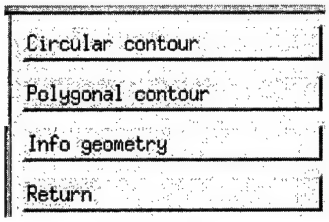

Figure 11: The « geometry of contour » edition menu

- **Electric nature** allows the user to define if the contour is a contrast (dielectric contour) or a frontier (metallic contour), with the following menu (Figure 12, menu number 77):
  1. **Metal**
  2. **Dielectric**
  3. **Info geometry**
  4. **Return**

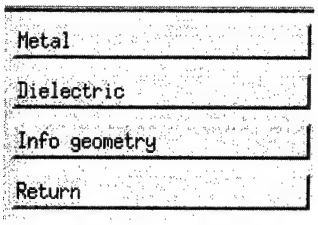

Figure 12: The « electric nature of contour » edition menu

- If the **Metal** option is chosen, the user is first asked which side of the contour is the metal side with the following question: ' ***Metallic side (L) or right R) ? [L]***, where the default value is Left. Then, the user will be plugged onto the following menu to define the left and right materials:

  1. **Left material**
  2. **Right material**
  3. **Info materials**
  4. **Info geometry**
  5. **Return'**

     - Depending which side of the contour is metallic, a check will be made on the material type indicated by the user. For instance, if the metal side is the left one, the left material will have to be of metallic nature and the right material will have to be of dielectric nature. If any error occurs, the characterization of the electric nature is not made.

- If the **Dielectric** option is chosen, the user will be plugged into the following menu in order to define the left and right materials:

  1. **Left material**
  2. **Right material**
  3. **Info materials**
  4. **Info geometry**
  5. **Return**

     - None of the left and right materials have to be of metallic nature. If any error occurs, the characterization of the electric nature is not made.

- **Info on materials** displays an information window on all the available materials,
- **Info on geometry** displays an information window on the current geometry being created.

- **Info on materials** is the same material information window appearing in several menus.
- **Info on geometry**: same menu used in all LAPLACE vertical menus,
- **Return**: goes back to the previous menu. If the nature of the contour (metallic or dielectric) is not specified, the user is asked if he wants to return. If the nature is not specified, the contour cannot be displayed in the window.

3.3.3.1 The "Circular contour" sub-menu

This menu proposes the following options (menu number 63):

1. **Center**
2. **Radius**
3. **Nb basis functions**
4. **Info on geometry**
5. **Return**

- **Center**: the user goes to the menu of selection of contours:
  - If he chooses the « With keyboard option » he has to enter the x and y positions of the center (real number in MKSA units),
  - If he chooses the « with mouse option » he has to specify the position of the center with a mouse-click in the window,
- **Radius**: the user has to enter the radius (real number in MKSA units) of the circle
- **Number of basis functions**: the user has to enter the number of basis functions to be applied on the circular contour (integer),
- **Return**: goes back to the previous menu

All those information items are not fully required in the description of the geometry. They can be saved incompletely in the « .geo » file in order to be completed later in another session of LAPLACE. However if they remain incomplete at a calculation stage, this calculation will be stopped.

3.3.3.2 The "Polygonal contour" sub-menu

First the program asks the user to enter the number of segments constituting the polygonal contour. A warning message is then entered to remind the user that the segments have to be entered counter clockwise.

Then, the parameters of segments are entered sequentially through the following menu (menu number 64):

1. **X Y of origin**
2. **Nb of unknowns**
3. **Repart. of unknowns**
4. **Next segment**
5. **Info on geometry**
6. **Exit polyg. contour**

- **XY segment-origin**: the user enters the X (real) and Y (real) coordinates of the origin of the segment,
- **Nb of unknowns**: the user enters the number of unknowns of the segment (integer). Entering a number of unknowns equal to zero is permitted and makes the contour virtually opened at the levels of this segment.
- **Repart. of unknowns**: the user indicates the repartition of the unknowns. For this he has to go into the following sub-menu:
  - **LIN:** linear repartition of unknowns from the origin,
  - **DEB conc. at origin**: concentration of the unknowns at the origin of the segment,

- **FIN conc. at end**: concentration of the unknowns at the remote extremity of the segment,
- **DEB then FIN**: first, concentration at the origin; then, at remote extremity of the segment,
- **Info on geometry**: same menu as in all the menus,
- **Return**: goes back to the previous menu.

- **Next segment:** with this button, the user goes to the next segment even if the three parameters mentioned before in the menu have not been entered. However, the user will be asked if he is sure to continue or not.
- **Info on geometry:** same menu as in all the menus,
- **Exit polyg. segment**: at any moment of the entering phase of the segment, the user may exit the menu. However, if all the segments have not been entered, the whole contour is deleted.

When the last segment is entered, a check is made to determine if the counter clockwise rule is respected. If not, the segments are automatically reoriented in order to have an appropriate contour direction.

3.3.4 The « modify contour » button

This button allows the modification of an existing contour. The first information asked for is the name of the contour. If the name of the contour does not exist, the user can create this contour through the "Create contour" button. Once the name of the contour accepted, the user enters the same menu as the "**Create contour**" menu.

3.3.5 The « Delete contour » button

This button allows the user to delete an existing contour and remove it of memory. The user has to specify the name of an existing contour. If this contour is the pole of associated circuits, those circuits are deleted at the same time the contour is deleted.

3.3.6 The « Move contour » button

This button allows the user to move a contour. If the contour is a circle, the center of the circle is re-positioned. If the contour is a polygon, the barycenter of this polygon is positioned. A click on this button launches the "Keyboard/Mouse" selection menu (Figure 6) for which the user can use a mouse or a keyboard action.

3.3.7 The « Edit group » button

This button allows the user to edit groups of contours. A click on this button launches the following menu (menu number 71, Figure 13):

1. **Create group**
2. **Modify group**
3. **Delete group**
4. **Info groups**
5. **Info geometry**
6. **Return**

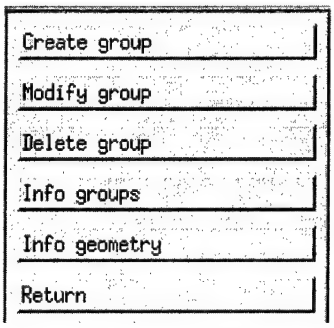

Figure 13: The « group » edition menu

- **Create group** and **Modify group** allow the user to create or modify a group by adding / removing groups. Prior to any action, the user is asked to enter the name of a group. For the creation, the name of the group has to be the name of new group. For the modification of a group, the name of the group has to be the name of an existing group. A click on one of the two buttons launches the following menu (menu number 72, Figure 14):
  1. **Add contour**
  2. **Remove contour**
  3. **Info geometry**
  4. **Return**

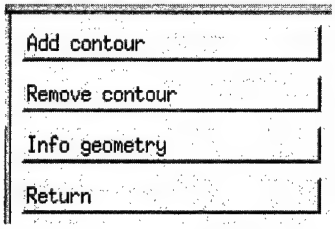

Figure 14: The « group contour » edition menu

- **Delete group** allows the user to delete an existing group. Prior to any action, the user is asked the name of the existing group to be deleted,
- **Information on groups** displays an information window on all the existing groups
- **Information on geometry** is the usual information window
- **Return** goes back to the "Group edition" menu.

3.3.8 The « Move group » button

This button allows the user to move groups of contours. If the contour is a polygon, the barycentre of the contours in the group is first determined. A click on this button launches the "Keyboard/Mouse" selection menu (Figure 6) for which the user can use a mouse or a keyboard actions.

3.3.9 The « Macro-Models » button

This button allows the user to edit specific groups of contours, related to generic geometries. The generic geometries considered in the software are the ones currently needed by the user when he wants to build a complex cable bundle geometry. The menu offers the following choices (menu number 73, Figure 15):

1. **Metallic plane**
2. **Coated wire**
3. **2-wire cable**
4. **3-wire cable**
5. **Random cable**
6. **Info geometry**
7. **Return**

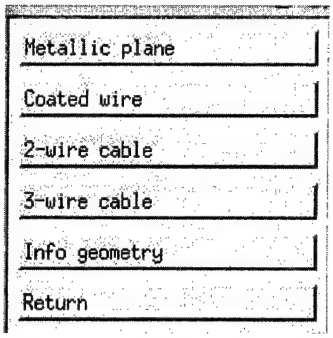

Figure 15: The « macro-model » edition menu

- **Metallic plane** allows the user to create a finite dimension ground plane. The ground plane is defined by two points, one being its middle, one being one of its extremities. The menu automatically adjusts to create the ground plane geometry,
- **Coated wire, 2-wire cable and 3-wire cable** stand for defining geometries of wires having an insulating dielectric. In this menu, all the wires are supposed to have the same characteristics. With a click on one these 3 buttons, the following menu is launched (menu number 80, Figure 16):
  1. **Centre**
  2. **Radius conductor**
  3. **Rayon insulator**
  4. **Conductor material**
  5. **Insulator material**
  6. **Nb basis fns cond**
  7. **Nb basis fns diel**
  8. **Info materials**
  9. **Info geometry**
  10. **Return**
  - **Centre** defines the coordinates of the centre (barycentre) of the group.
  - **Radius conductor** defines the radius of the metal contour of the wire.
  - **Radius insulator** defines the radius of the insulating dielectric.
  - **Conductor material** defines the material to be affected to the conductor contour (left material). The right contour is automatically affected with the material characterizing the dielectric insulator.
  - **Insulator material** defines the material to be affected to the dialectic contour (left material). The right contour is automatically affected with the "medium" material.
  - **Nb basis functions conductor** determines the number of basis functions of the conductor contour. A typical value of 20 is generally appropriate in most configurations.

- **Nb basis functions dielectric** determines the number of basis functions of the dielectric contour. A typical value of 30 is generally appropriate in most configurations.
- **Information on materials** displays the usual information window on the existing materials.
- **Information on geometry** displays the usual information window on the current geometry.
- Return goes back to the previous "Macro-model" edition menu.

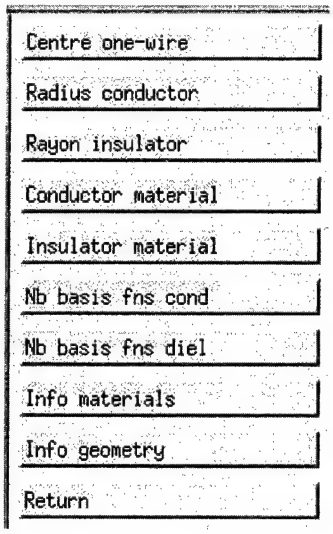

Figure 16: « Group contour characteristic » edition menu

- **Random cable** allows the user to define a random geometry of N cables bundle. With a click on this button, the user goes in the following menu:
  1. **From the interface**
  2. **From a file**
  3. **Random type**
  4. **Return**
  - **From the interface** will plug the user with the « Group contour » characteristic menu already seen in Figure 16 for the one wire, 2-wire cable and 3-wire cable buttons (all the characteristics being the same for all the conductors).
  - **From a file**, the reader will define its random geometry by reading a file. With this option, the characteristics of the wires may be different. A click on this button will lead the user to the following menu):
    1. **Enter file name**
    2. **File format**
    3. **Return**
       - **Enter file name**: the user has to give the name of a file name previously entered by hand with a text editor by the user,
       - **File format** reminds the user the format of the file with an information window. The format of the file is the following:
       - **Return** goes back to the previous menu.

- **Random types**, provides the user 2 types of random cable geometry generations reported in the following menu:
  1. **Full random**
  2. **Partially random**
  3. **Return**
- **Full random** allows the user to generate a geometry that will **never** be the same each types he addresses this feature,
- **Partially random** allows the user to generate a geometry that will **always** be the same each types he addresses this feature,
- **Return** goes back to the previous menu and generates the random geometry.

- **Information on geometry** is the usual information window
- **Return** goes back to the "Group edition" menu.

3.3.10 The « Create circuits » button

This button is for creating circuits on the currently available geometry. Two manners of creating circuits are offered to the user through the following sub-menu (Figure 17):

- **One-reference circuit**: this button offers the possibility to create circuits having the same minus pole. There will be as many circuits as there are conductors (except the conductor corresponding to the specified minus pole). If circuits already exist, they will be reset and replaced by the new circuits.
- **One circuit**: this button offers the possibility to create a circuit by entering the names of its plus and minus poles (Figure 18):
  - **Enter + pole**: the user gives the name of the contour corresponding to the plus pole. This name cannot be the same name as the name of the minus pole,
  - **Enter - pole**: the user gives the name of the contour corresponding to the minus pole. This name cannot be the same as the name of the plus pole,
  - **Info geometry**: same menu used in all LAPLACE's vertical menus,
  - **Return**: goes back to the previous menu.
- **Info on geometry**: same menu used in all LAPLACE vertical menus,
- **Return**: goes back to the previous menu.

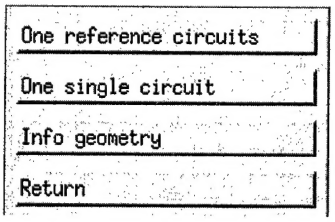

Figure 17: The « create circuit » menu

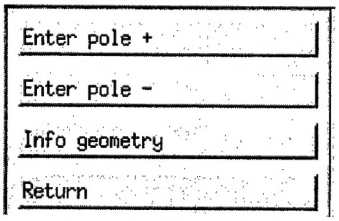

Figure 18: The « enter poles » menu

3.3.11 The « Delete circuit » button

This button is for deleting existing circuits on the currently available geometry. Two buttons are offered to make a circuit destruction operation in the following sub menu:

- **One circuit**: the user has to enter the name of the circuit,
- **Circuits from a contour**: the user has to specify the name of an existing contour. All the circuits for which this contour is a minus or a plus pole are deleted,
- **Info on geometry**: same menu used in all LAPLACE's vertical menus,
- **Return**: goes back to the previous menu

3.3.12 The « Save geometry » button

This button is for saving the current geometry information available in memory in a « .geo » file. The user has to enter the name of the file through the file widget.

At this stage other checks are made on the parameters to avoid the largest number of errors.

3.4 The "Calculation" menu

3.4.1 Preliminary checks

First checks are made on the dimension of tables allowed in the compiled version of the program:

- number of contours. Typical value is maxcontour=140.
- number of segments per polygonal contours. Typical value is 40.
- number of basis functions per circular contours.

Before any calculation, preliminary checks on the consistency of the parameters are also made when reading the « .geo » file:

- radius of circular contours not equal to zero,
- number of basis functions of a circular contour not equal zero,
- number of segments not equal to zero on a segment of a polygonal contour,
- total number of segments not equal to zero,
- no left or right permittivity equal to zero for a dielectric contour (contrast),
- left or right hand side permittivity, but not both, equal to zero for a conductor (frontier),
- number of circuits equal to the number of conductors minus 1,
- existence of the name of poles,
- no similarity between minus and plus poles,
- all values not negative.

If one of those checks reveals an error, those errors are listed and the calculation is not launched. The user has to go back to the "**Create geometry**" menu to enter appropriate parameters.

No check is made on the fact that the contours may touch each other.

3.4.2 Characterization of the calculation

The calculation goes through the following menu:

- **Input file**: this button specifies the name of the input « .geo » file,
- **Calculation options**: this button gives the user several options for his calculation:
  - **L and C**: this options makes the classical calculation of L and C values
    - **Free space only**: the user indicates if he wants to do the free space calculation only (this one is always done),
    - **Averaging of the L and C parameters**: for some applications, it may be interesting to average the L and C parameters. Several options are offered:
      - **Circulant matrices**
      - **Twisted cables**: this option averages the diagonal terms and the off-diagonal terms
    - **Electric Potential**: this option generates the calculation of the potentials around each conductor. The information is saved in a data file that can be displayed with any 2D viewer later on.
  - **Resistance**: this options makes the calculation of the per unit length resistance value (DC plus frequency dependent part)
  - **Conductance**: this options makes the calculation of the per unit length conductance matrix
  - **Info on geometry**: same menu used in all LAPLACE vertical menus,
  - **Return**
- **Output file**: this button specifies the root of the output file. Automatically, the following extensions are given:
  - « **.tub** », for the ASCII file containing the per unit length parameters of the actual geometry (dielectric included) if the result is available
  - « **.t** » , for the direct access file containing the per unit length parameters of the actual geometry (dielectric included) if the result is available
  - « **_fs.tub** », for the ASCII file, if the free space calculation option is activated
  - « **_fs.t** », for the access direct file, if the free space option is activated (to be implemented)
  - « **_.pot** », for the ASCII file containing the electric potential distribution.
- **Info on calculation**: the button allows the user to check the parameters of its geometry before launching the calculation,
- **Launch calculation**: the button launches the calculation. The window displays the main operations realized by the code.
- **Info on geometry**: same menu used in all LAPLACE's vertical menus,
- **Return**: goes back to the main menu.

4 SEE ALSO

CABLE-BLINDE(1), CREER-TUBE(1), FORMAT-TUBE(1), VISU-TUBE(1)